



TypeScript cheat sheet



```
const item = Scene.getItem("myItem");  
item.transform.rotateBy(0, 180, 0, 1); // Rotate by 180 degrees in 1 second
```

```
const item = Scene.getItem("myItem");  
item.input.onClick(() => {  
  // Action to perform when clicked  
});
```

```
const music = Sound.load("soundID");  
music.play(); // Play the sound
```

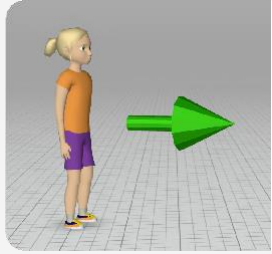
```
GUI.HUD.showChoicePanel({  
  question: 'This is your question.',  
  answer1: 'first answer.',  
  answer2: 'second answer.',  
  onAnswer: answer => {  
    if (answer === 1) {  
      // Action for first answer  
    }  
    else if (answer === 2) {  
      // Action for second answer  
    }  
  }  
});
```

Last update: September 2024

**Quick descriptions of
the beginners TypeScript manual**

Transform

Action to perform

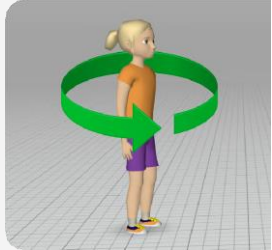


Move an Object or Character over time

TypeScript code to use

```
const item = Scene.getItem("myItem");
item.transition.moveBy(new Vector3(1, 0, 0), 1);
```

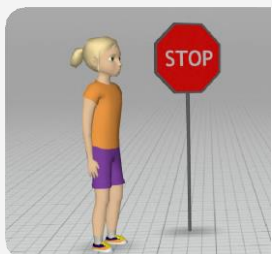
Move item by 1 meter on the x-axis in 1 second.



Local Rotation of Object over time

```
const item = Scene.getItem("myItem");
item.transition.rotateLocal(new Vector3(0, 0, 1), 3, 1);
```

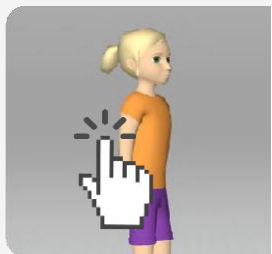
Rotate item locally by 3 degrees on the z-axis in 1 second.



Stop an object or a character

```
item.transition.stop(); // Stops the current transition of the object
```

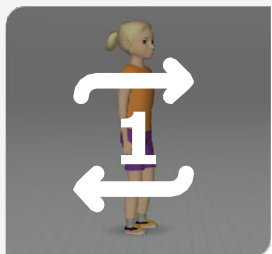
Events



Make something happen when an item is clicked

```
const item = Scene.getItem("myItem");
item.input.onClick(() => {
  // Action to perform when clicked
});
```

Control



Make one or more actions **repeat** a certain amount of times

```
for (let i = 0; i < X; i++) {
  // Repeated action
}
```

Repeats the action x times



Make my program **wait** for some time

```
Time.schedule(() => {
  // Action after 1 second
}, 1);
```

Performs the action after 1 second.



Go to specific scene.

```
Space.goToScene(1);
```

Switches to scene with ID 1.

Actions

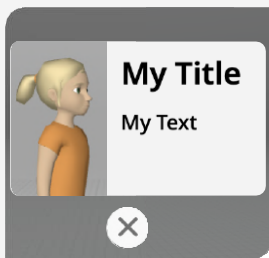
Action to perform



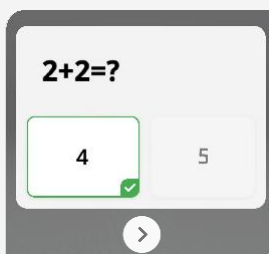
Make a character **say** or **think** something



Change the **opacity** of an item



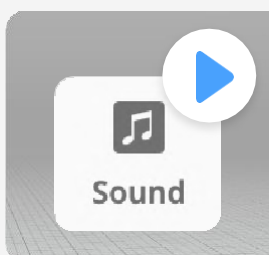
Show an **info panel** with a title, text (and an image)



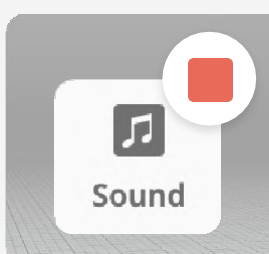
Show a **quiz panel** with a question and answers that can be clicked and will trigger an action when the selected answer is correct or incorrect.



Show a **choice panel** with a question and options that can be clicked and that trigger different actions.



Play a **sound** file



Stop playing the **sound** file

TypeScript to use

```
const character = Scene.getItem("myCharacter");
character.speech = "Hi!"; // Character says "Hi!"

character.thought = "Hmm"; // Character thinks "Hmm"
```

```
const item = Scene.getItem("myItem");
item.opacity = 0.5; // Set the opacity to 50%
```

Sets the opacity(see-through level) of the object to 50%.

```
GUI.HUD.showInfoPanel({
  title: "Title",
  text: "Text",
  image: "imageUrl" // Optional: Add image URL or leave it out for no image
});
```

```
GUI.HUD.showQuizPanel({
  question: 'Your question here',
  answer1: 'Answer 1',
  answer2: 'Answer 2',
  answer3: 'Answer 3',
  answer4: 'Answer 4',
  correctAnswer: 1, // Change to the correct answer number
  onCorrect: () => {
    // Action to perform on correct answer
  },
  onWrong: () => {
    // Action to perform on wrong answer
  }
});
```

```
GUI.HUD.showChoicePanel({
  question: 'This is your question.',
  answer1: 'First answer.',
  answer2: 'Second answer.',
  onAnswer: answer => {
    if (answer === 1) {
      // Action for first answer
    }
    else if (answer === 2) {
      // Action for second answer
    }
  }
});
```

```
const music = Sound.load("soundID");
music.play(); // Play the sound
```

```
music.stop(); // Stop the sound
```